TRANSPORTATION ANALYSIS SIMULATION SYSTEM (TRANSIMS)

Version: **TRANSIMS-LANL-1.0**

VOLUME 2 – SOFTWARE
PART 3 – TEST NETWORKS

28 May 1999

LA-UR 99-2576

TRANSIMS

Version: TRANSIMS-LANL-1.0

VOLUME 2 – SOFTWARE
PART 3 – TEST NETWORKS

28 May 1999

LA-UR-99-2576

The following persons contributed to this document:
C. L. Barrett*
R. J. Beckman*
K. P. Berkbigler*
K. R. Bisset*
B. W. Bush*
S. Eubank*
J. M. Hurford*
G. Konjevod*
D. A. Kubicek*
M. V. Marathe*
J. D. Morgeson*
M. Rickert*
P. R. Romero*
L. L. Smith*
M. P. Speckman**
P. L. Speckman**
P. E. Stretz*
G. L. Thayer*
M. D. Williams*

* Los Alamos National Laboratory, Los Alamos, NM 87545
** National Institute of Statistical Sciences, Research Triangle Park, NC

Acknowledgments

This work was supported by the U. S. Department of Transportation (Assistant Secretary for Transportation Policy, Federal Highway Administration, Federal Transit Administration), the U. S. Environmental Protection Agency, and the U. S. Department of Energy as part of the Travel Model Improvement Program.

CONTENTS

PREFACE 6

1. REPRESENTATION OF TRANSPORTATION NETWORKS 7

1.1 NODES AND LINKS 7
1.2 LANES.. 9
1.3 PARKING PLACES AND TRANSIT STOPS.. 11
1.4 ACTIVITY LOCATIONS AND PROCESS LINKS.. 11
1.5 TRAFFIC CONTROLS 12
1.6 STUDY AREAS... 14

2. CIRCLE NETWORK 15

2.1 OVERVIEW... 15
2.2 DESCRIPTION 15
2.3 USAGE 17

3. TEE NETWORK...21

3.1 OVERVIEW... 21
3.2 DESCRIPTION 21
3.3 USAGE 22

4. MULTIMODE NETWORK..25

4.1 OVERVIEW... 25
4.2 DESCRIPTION 25
4.3 USAGE 27

5. LOCAL STREETS NETWORK 29

5.1 OVERVIEW... 29
5.2 DESCRIPTION 29
5.3 USAGE 32

Preface

Multiple roadway, transit, and walk networks are given to demonstrate various aspects of the TRANSIMS framework. The first two networks, a one- or three-lane circle with merge and left turn lanes and a tee ("T")with a traffic signal, are used to calibrate and test the Traffic Microsimulator. Activities are not generated for these networks, and plan sets are generated using specialized codes—not the more general Route Planner.

Two networks are available which exercise the TRANSIMS modules and the TRANSIMS framework: the Multimode Network and the Local Streets Network. The Multimode Network allows for walking, driving and riding the bus. Specialized, simplified population and activity generators are given to produce populations and activities for this network. With these activities, all modules of TRANSIMS—including feedback using selector scripts and multimode routes—are useable.

The full-feed forward portion of TRANSIMS using the actual Population Synthesizer and Activity Generator can be exercised using the Local Streets Network. Populations for this network are created and distributed using the Population Synthesizer. Activities are produced for the individuals in the population using the full TRANSIMS Activity Generator. Feedback to the activity list given by the Activity Generator using the Selector scripts is not available at this time.

1. REPRESENTATION OF TRANSPORTATION NETWORKS

A TRANSIMS network provides a high-fidelity representation of the transportation infrastructure within an urban area. It represents the configuration of lanes on streets and highways, the signage and signals controlling traffic, and the parking and transit facilities adjacent to the roadway. TRANSIMS models the various network entities with the objects discussed in this section: nodes, links, lanes, pocket lanes, parking places, transit stops, activity locations, process links, unsignalized controls, signalized controls, detectors, and signal coordinators. A set of 19 relational-format data files provide the external specification for these objects (see Volume 3—*Files*).

1.1 Nodes and Links

Nodes and links form the basic *skeleton* of the transportation network. A *link* is the part of the network corresponding to an edge in graph theory. Links represent street, road, highway, rail, and sidewalk segments. Each link has a constant number of permanent lanes, but may have a variable number of pocket lanes. A link may have lanes in both directions; alternately, the lanes in opposite directions may lie on separate links (in which case no passing into oncoming lanes is possible). Each link has a unique ID, a street name, a length, setback distances for limit lines that might be present at its ends, lanes for traffic, speed limits, a functional class, and vehicle type restrictions. Table 1 defines the functional classes used for links.

Table 1. Functional class definitions for roadway links in TRANSIMS.

Name	Interpretation
Freeway	A divided, arterial highway for through traffic with full control of access. Full access control means the authority to control access is exercised to give preference to through traffic by providing access connections with selected public roads, but prohibiting grade crossings and/or direct private driveway connections.
Expressway	A divided, arterial highway for through traffic with partial control of access. Partial control of access means that some authority is exercised to control access in the manner described above, but there are crossings at grade and/or direct private driveway connections.
Primary Arterial	A major arterial roadway with intersections at grade crossings and direct access to abutting property and on which geometric design and traffic-control measures are used to expedite safe movement of through traffic.
Secondary Arterial	A minor arterial roadway with intersections at grade crossings and direct access to abutting property and on which geometric design and traffic-control measures are used to expedite safe movement of through traffic.
Frontage Road	An arterial that runs parallel to a freeway or expressway.
Collector Street	A roadway on which vehicular traffic is given preferential right of way, and at the entrances to which vehicular traffic from intersecting roadways is required by law to yield right-of-way to vehicles on such a roadway in obedience to either a stop sign or a yield sign, when such signs are erected.
Local Street	A street or road primarily for access to residence, business, or other abutting property.
Freeway Ramp	A unidirectional roadway providing connection between a freeway or expressway and an arterial.

Name	Interpretation
Zonal Connector	An imaginary (non-physical) connection to or from the centroid of a traffic analysis zone.
Other	Any roadway not fitting the above definitions.
Walkway	A street restricted to use by pedestrians.
Busway	A street restricted to use by buses.
Light Rail	A roadbed restricted to use by light rail cars.
Heavy Rail	A roadbed restricted to use by heavy rail cars.
Ferry	A waterway crossed by ferry.

A *node* is the part of the network corresponding to a vertex in graph theory. Nodes typically occur at intersections in the road network. A node must exist where the network branches and where the permanent number of lanes changes, but a node may also exist anywhere along a street for purposes of convenience. Nodes are not required where turn pockets start or end because these are not considered permanent lanes. (A node may be present where either of the aforementioned occurs, however.) Figure 1 illustrates some node placements—the nodes in this diagram exist for the following reasons:

- Nodes 1 and 5 exist because there are intersections at these locations in the network between links 1, 2, 3, and 5, and between links 7, 8, 9, and 10, respectively.
- Node 2 exists because link 3 and link 4 have a different number of permanent lanes (i.e., permanent lanes merge at this point).
- Node 3 exists because links 5 and 6 have different street names (i.e., the link attributes change here).
- Node 4 exists because the original data source for this network has a node in this location; none of the link attributes are different between links 6 and 7. This node is not strictly necessary to represent the network—it only exists for the analyst’s convenience.
- No node exists between links 9 and 11 because the former passes over the latter (i.e., they do not connect).

Each TRANSIMS node has a unique ID and a location in three-dimensional space (easting and northing in UTM coordinates and an elevation in meters). Each node has a traffic control associated with it (signed, pre-timed, actuated, coordinated, etc.).

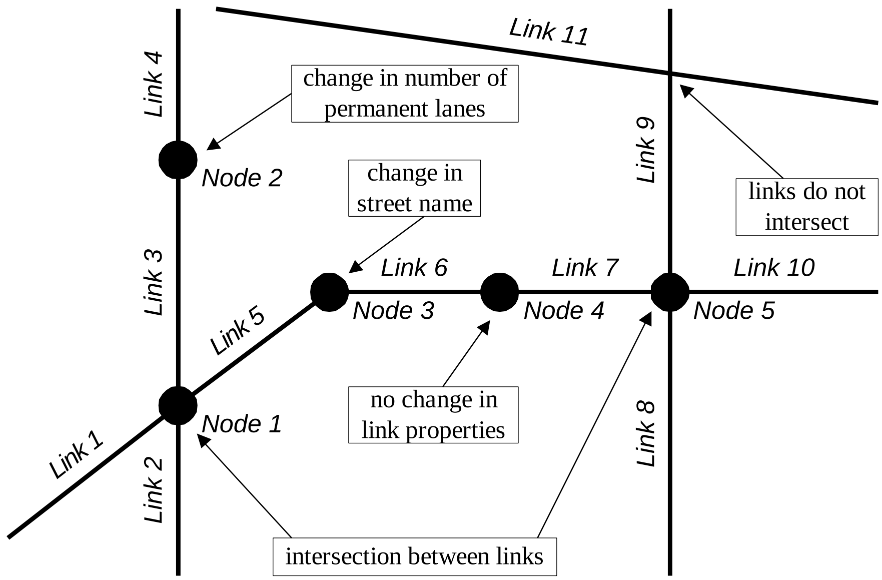

Figure 1. Illustratrative relationships between nodes and links.

1.2 Lanes

A *lane* is where traffic flows on a link. A *pocket lane* is one of the following:

- a right- or left-turn pocket (a lane that starts after the "from" node and ends at the "to" node),
- a right or left pull-out (a lane that starts after the "from" node and ends before the "to" node), or
- a right or left merge pocket (a lane that starts at the "from" node and ends before the "to" node).

If a lane starts at the "from" node and ends at the "to" node, it is considered a permanent lane, not a pocket lane.

The lanes on each side/direction of the link are numbered separately, starting with lane number one as the leftmost lane (relative to the direction of travel). Each successive lane to the right of it is numbered one greater than its predecessor. A lane is considered permanent if it is not a temporary pocket lane. Pocket lanes (i.e., turn pockets, merges, and pull-outs) are numbered in sequence—even if they do not exist for the full length of the link. A two-way left-turn lane, if present, is considered to be lane number zero. Figure 2 illustrates how lanes are located and numbered in practice—note especially,

- the numbering of the pocket lanes and two-way turn lane on link 12384,
- the presence of a pull-out lane on link 12407,
- the presence of merge lanes on link 12384, and
- the presence of a turn pocket lane on link 28804.

Each lane on a link may have its own speed limit (as a function of vehicle type) and vehicle restriction—this is also illustrated in Figure 2.

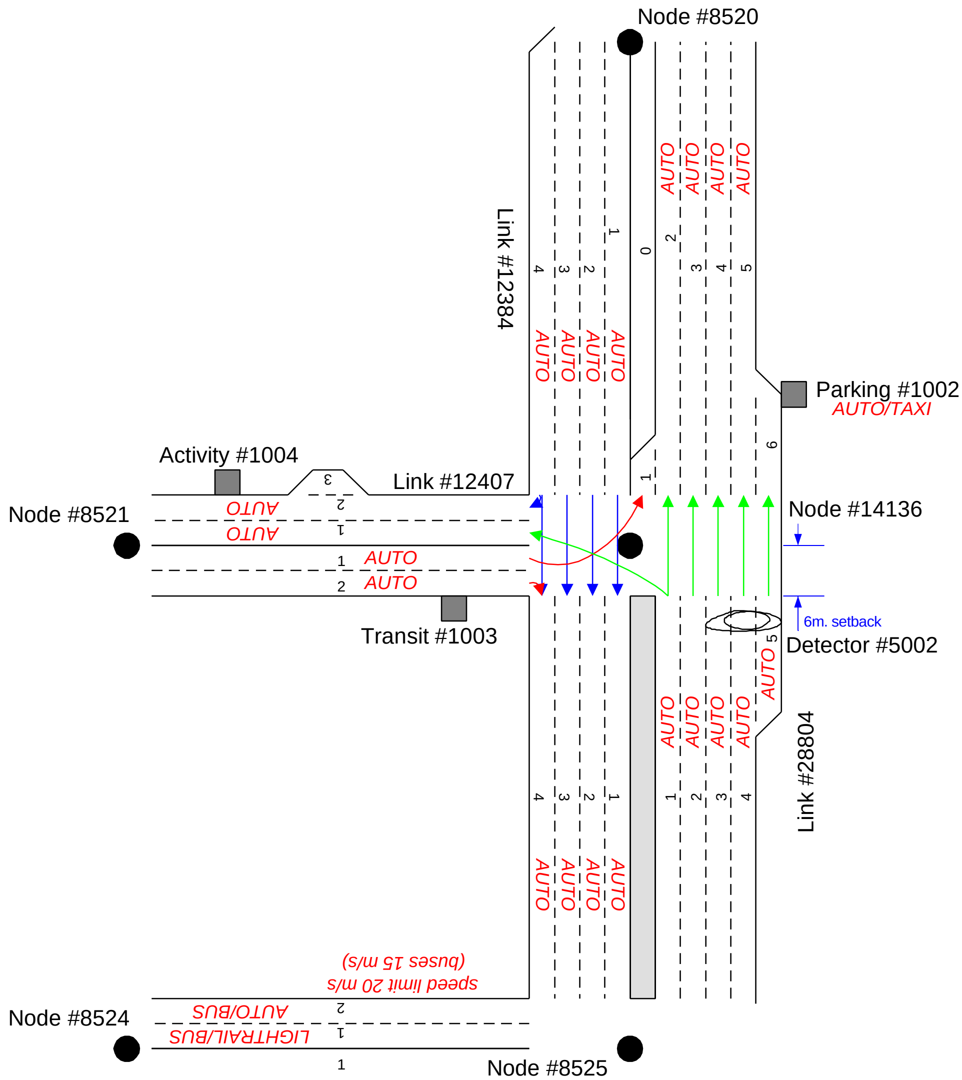

Figure 2. Illustrative relationships between lanes and other network objects.

Lane connectivity specifies how lanes are connected across a node. Incoming and outgoing links and lanes are defined relative to the node. For each incoming lane on an incoming link, at least one outgoing lane must be specified for each outgoing link that a vehicle on the incoming link can transition to. Multiple outgoing lanes may be defined for an outgoing link, if desired. The colored arrows at node 14136 in Figure 2 illustrate allowed movements for that intersection.

1.3 Parking Places and Transit Stops

Parking places, and transit stops are known in TRANSIMS terminology as *link accessories*. Link accessories have a *sense of direction* indicating on which side of the link they reside. Parking places and transit stops are points on a link where vehicles reside when they are not traveling on the network, and where travelers change modes. Figure 2 illustrates the locations of several link accessories.

Parking places are located along links and are used as origins and destinations for vehicle trips. Parking may be placed where it is physically located in the network, or it may be placed in aggregate generic parking areas representing several of the driveways, lots, parking places, etc., on a link. Parking areas have a *style* (parallel on-street, head-in on-street, driveway, lot, or generic), and vehicle capacity and vehicle type restrictions.

A *transit stop* is a location on a link where a transit vehicle, such as a bus or light rail car, waits to embark and disembark passengers. Transit stops have a *style* (station, stop, or yard), and vehicle capacity and vehicle type restrictions.

1.4 Activity Locations and Process Links

Activity locations and process links supplement the representation of the physical network with information about the logical "network" confronting TRANSIMS travelers. Figure 3 illustrates how the physical transportation network is divided into logical layers corresponding to different modes of travel (e.g., walking, driving a car, or riding a bus). Travelers experience mode changes when they move from one level to another; vehicles are restricted to moving on a particular level, however.

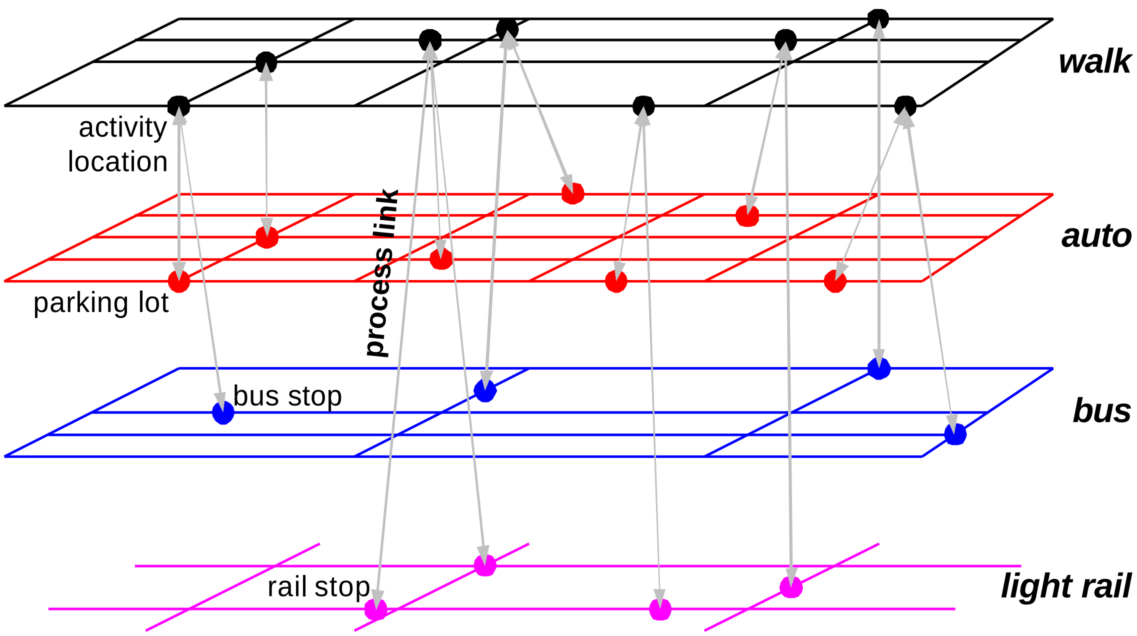

Figure 3. Illustration of logical network layers used in planning traveler activities.

An *activity location* is a place on a link (i.e., a link accessory) where traveler activities (such as work, home, or shopping) can take place. Activity locations typically reside on the "walk" network, as in Figure 3, but they might be located on other layers if a specific transportation study requires that. Attributes such as three-dimensional location (easting and northing in UTM coordinates and elevation in meters) are also attached to activity locations, along with land use (amount of residential/commercial/industrial land in the nearby area, employment statistics, accessibility to transit, etc.). The illustration in Figure 2 contains an activity location on link 12407.

A *process link* is a "virtual" connection between an activity location, parking place, or transit stop and another activity location, parking location, or transit stop. It represents the process of a traveler changing modes and accounts for the cost (in time and money) for the traveler to make the mode change; it does not represent a physical segment of road or sidewalk. Figure 4 shows how these process links represent these costs.

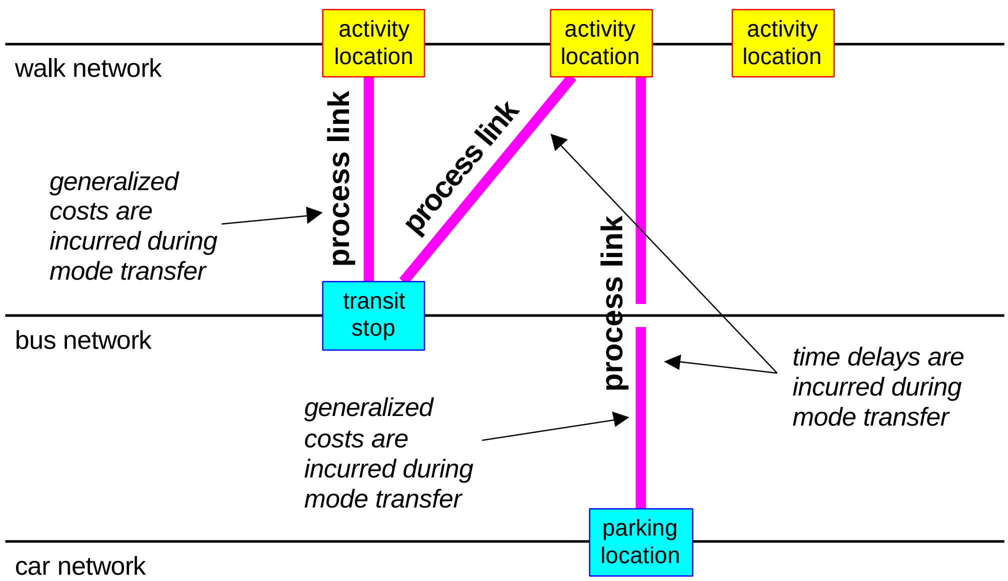

Figure 4. Illustration of process links connecting walk, bus, and car modal network layers.

1.5 Traffic Controls

Each node has a traffic control associated with it. A *traffic control* specifies how vehicles can move across lanes connected at a node and who has the right-of-way. Controls may be either unsignalized (i.e., signed) or signalized. Intersections typically have a spatial extent determined by the link setback distances at the node (see Figure 2).

An *unsignalized control* represents the type of sign control, if any, that is present at an unsignalized node. Nodes where only the number of permanent lanes is changing (i.e., nodes used to represent merging of permanent lanes) are generally considered unsignalized. Each link with incoming lanes at an unsignalized node has either a stop sign, a yield sign, or no sign specified for it.

A *signalized control* represents a traffic light. Each signal must have a timing plan and a phasing plan. Signals may be pre-timed or actuated. Actuated controllers may be single- or dual-ring, and dual-ring controllers may have barriers and be single- or dual-entry (see Figure 5); timing and phasing plans may change throughout the day. A *timing plan* specifies the lengths of the intervals (minimum green, maximum green, green extension, yellow, and red clearance times) during the specific phases for a traffic light and how the phases are grouped and ordered in rings. Many nodes in a network may share the same timing plan. It is possible for each phase to transition to more than one phase if required, or a phase may be skipped occasionally. A *phasing plan* specifies the turn protection (protected, unprotected, or unprotected after stop) in effect for transitioning from an incoming link to an outgoing link during a particular phase of a specific timing plan: a vehicle making a *protected* movement need not look for oncoming or interfering traffic whereas a vehicle making an *unprotected* movement must check for oncoming and interfering traffic when executing the movement.

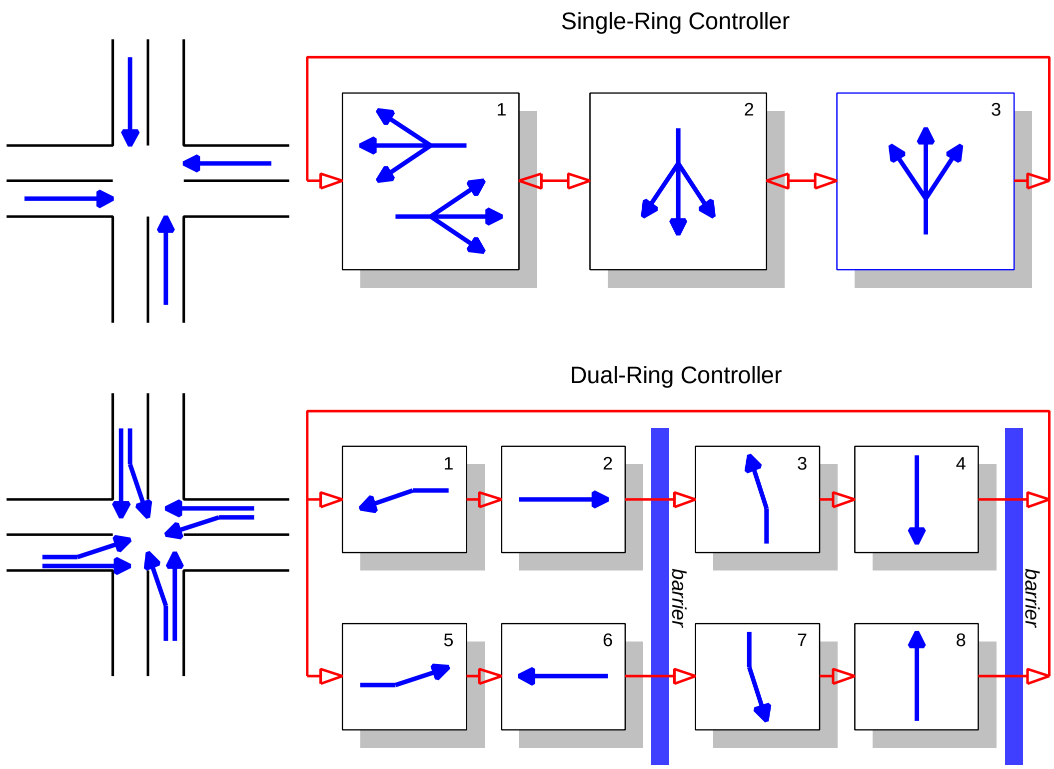

Figure 5. Single- and dual-ring signalized traffic control configurations.

A *detector* is a device that identifies the presence or passage of a vehicle over an area of the lanes on a link; these are used as the triggers for actuated controllers. The area covered by a detector is defined in terms of the segment of lanes covered; detectors may register either the presence of vehicles in the area or the passage of vehicles through the area. Figure 2 shows a detector on link 28804. A *signal coordinator* is a device that controls the operation of one or more traffic controls; it uses its *coordination algorithm* to coordinate the work of several traffic controls using detector input.

1.6 Study Areas

The microsimulation distinguishes two types of links in its calculations: *Study area links* are the links of interest for the traffic analyst. The output subsystem, for instance, records events such as when a vehicle leaves or enters the study area. The nature of the microsimulation algorithms makes it necessary to simulate traffic on additional *buffer area links*. Typically, these links form a fringe about two links thick around the study area. A simulation includes buffer links in order to avoid edge effects such as when vehicles enter the study area on its boundary; the buffer gives these vehicles time to interact with other traffic and achieve realistic behavior before entering the study area.

2. CIRCLE NETWORK

2.1 Overview

The Circle Network, with merge and turn lanes, is used to calibrate and test the microsimulation. There are two circle networks—one with one lane and another with three lanes. Multiple calibrations are run on these basic networks. First, freeway traffic is calibrated by looking at traffic—uninterrupted by turns or merges—around the one- and three-lane circles. The one-lane circle demonstrates car following behavior, while the dynamics of lane changing is apparent using the three-lane circle. The speed limit on the circle is 37.5 meters per second (five cells per second) for the Freeway calibrations and 22.5 meters per second (three cells per second) for both the merge and left-turn calibration networks.

Both the one- and three-lane circles have intersections (or nodes) where vehicles can merge into the traffic moving around the circle or cross the traffic on the circle. These are used to measure (and calibrate) the effects of gap acceptance parameters for left turns and merging.

2.2 Description

Both the one- and three-lane circle calibration networks have thirteen unidirectional links. Links 1 through 7 form the circle. These links and the order of their connections are shown in Figure 6. The link numbers are given in the rectangles surrounding the diagram. Traffic on the circle moves from link 1 to 2 to 3 etc., before returning to link 1 from link 7. As shown in Figure 6, the actual roadway is laid out as a square, but traffic behaves as if it were a circle. All nodes on this portion of the network act as *seamless* nodes. Vehicles do not pause or slowdown at the corners of the box. The maximum speed on these seven links is 37.5 meters per second (five *cells* per second).

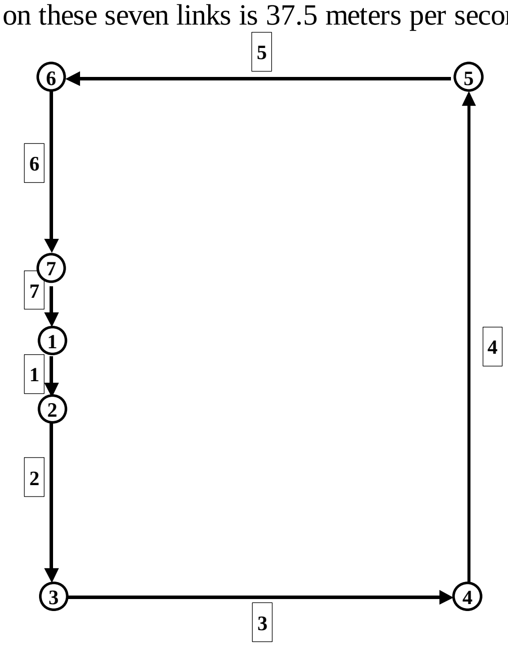

Figure 6: The structure and connections of the seven links and nodes making up the one- and three-lane circles.

Links 3, 4 and 5 are 1875 meters long. The shorter links 2 and 6 are 750 meters in length, while the shortest links 1 and 7 are but 187.5 meters. Links 8 through 12 allow for traffic to merge into the traffic on link 1 or cross the traffic at the node joining links 1 and 7, node 1. Figure 7 shows these links.

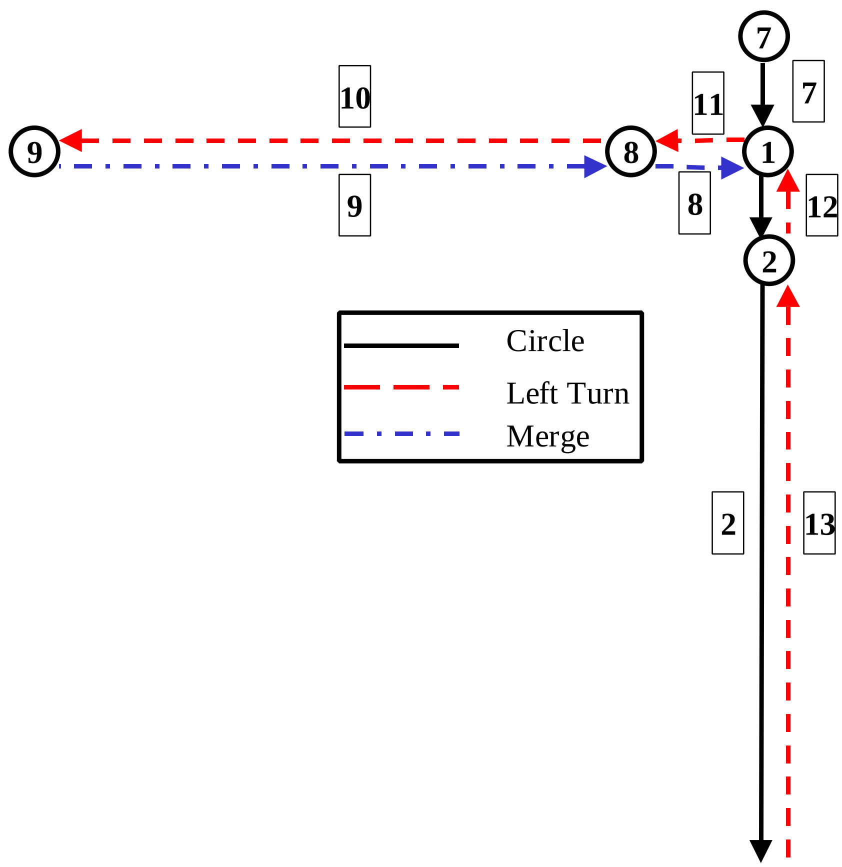

Figure 7: Links of the circles, left turn and merge networks are shown. Links 1, 2 and 7 are on the circle. Link 12 allows for left turns across the traffic at node 1. Link 8 permits merges into the traffic below link 7 on link 1.

For right turn or merge calibrations, traffic originates on link 9, moves to link 8, and merges into the circle traffic on link 1. Stop, yield, or no controls may be placed at the node to control traffic moving from link 8 to link 1.

To calibrate gap acceptance for left turns across traffic, traffic starts on link 13, moves to link 12, crosses the circle traffic at the node and continues on to links 11 and 10. Vehicles turning left must yield to the circle traffic. The circle traffic is not affected by the turning movements of the other vehicles.

2.3 Usage

The Circle Networks are used to calibrate the Traffic Microsimulator. Each is executed using a preset plan file, and the output is filtered to produce an analysis file containing the statistics of interest.

2.3.1 Calibration Filters

The filters on the freeway, merge, and left-turn calibration runs collect summary data on five-cell sample blocks placed at specified locations on the calibration networks (Table 2). The data is summarized over three-minute intervals. The sample blocks cut across all lanes of the link, but data is reported on a lane-by-lane basis, as well as the link totals.

Table 2: Calibration Network – Sampling Locations

Network	Sample Block Location	Sampled Type
Freeway	Sites 491 - 495 on the circle	Circle vehicles
Merge	Sites 491 - 495 (Link 7) on the circle Sites 501 - 505 (Link 1) on the circle	Circle vehicles Merging vehicles
Left Turn	Sites 491 - 495 (Link 7) on the circle Sites 1 - 5 (Link 11) on exiting link	Circle vehicles Left-turn vehicles

2.3.2 Freeway Calibrations

One- and three-lane freeway traffic is calibrated by moving traffic around the circle on links 1 to 7. Plans for these two calibrations are in files *$TRANSIMS_HOME/data/plans/Freeway1.plans* and *$TRANSIMS_HOME/data/plans/Freeway3.plans*. These files contain vehicle plans that start vehicles on one of the seven links and move the vehicles around the circle. The density of vehicles is continuously increased by adding more vehicles to the network. The vehicles continue around the circle and are not removed. A small snippet of one vehicle's plans is given below. In this plan, vehicle and individual number 1 starts at parking location 4 on link 4 then passes through nodes 5, 6, 7, 1, 2, 3 and 4 in order.

```
1 0 1 1 1 1
2 4 2 1 2
3360358 3360360 1
1 0 1
3361
1 0
5 6 7 1 2 3 4
5 6 7 1 2 3 4
5 6 7 1 2 3 4
5 6 7 1 2 3 4
```

The microsimulation for the two freeway calibrations is carried out using the configuration files *Freeway1_config* and *Freeway3_config* using the commands

```
% perl $TRANSIMS_HOME/scripts/Msim.pl $TRANSIMS_HOME/config/Freeway1_config sim.log
```

and

```
% perl $TRANSIMS_HOME/scripts/Msim.pl $TRANSIMS_HOME/config/Freeway3_config sim.log
```

Snapshot output is collected on link 7. This output is filtered using the program *$TRANSIMS_HOME/bin/FreewayFilter***.** This filter is executed by the command (where <> denotes input data)

```
% $TRANSIMS_HOME/bin/FreewayFilter <# lanes> <snapshot file name> <output file>
```

Example:

```
% $TRANSIMS_HOME/bin/FreewayFilter 1 $TRANSIMS_HOME/output/Freeway1/freeway1_snap.veh freeway.filtered
```

This produces output files with the format in Table 3.

Table 3: Freeway filter format.

Field	Interpretation
Simulation Time	Seconds since simulation start.
Lane	Lane number.
Density	Vehicles/km/lane at the sample block on the circle.
Flow	Vehicles/hour/lane at the sample block on the circle.

The columns contain the vehicle flows and roadway densities on the lanes on link 7. These can be plotted to produce a diagram showing the relationship between the flows and the densities.

2.3.3 Merge Calibrations

Merge behavior is studied using plans in the plan file *$TRANSIMS_HOME/data/plans/Merge2.plans***.** This file contains plans for vehicles traveling around the circle as in the freeway calibrations. Additional vehicles are given plans that start at parking location 13 on link 8 and merge into the traffic on link 1 at node 1. These vehicles proceed through nodes 2 and 3. They are removed from the network at the parking location 3 on link 3. A sample of these plans is given below.

```
1000001 0 1 1 1 1
0 13 2 3 2
8753 8753 1
1 0 1
5
1000001 0
1 2 3
```

The microsimulation for the merge calibration is carried out using the configuration file, *Merge2.config* using the command line

```
% perl $TRANSIMS_HOME/scripts/Msim.pl $TRANSIMS_HOME/config/Merge2_config sim.log
```

Snapshot output is collected on links 1 and 7 and is filtered for analysis using the program *$TRANSIMS_HOME/bin/MergeFilter* **.** The file produced by this filter contains the vehicle flow and density on link 7 and the flow of the merging vehicles on link 1. These data may be plotted to assess the merge rate as a function of the flow of oncoming traffic. The format of this file is shown in Table 4.

Table 4: Merge filter format.

Field	Interpretation
Simulation Time	Seconds since simulation start.
Lane	Lane number.
Flow-7	Flow of circle traffic in vehicles/hour/lane (Link 7).
Density-7	Density of circle traffic in vehicles/km/lane (Link 7).
Flow-1	Flow of merging vehicles in vehicles/hour/lane (Link 1).

This filter is executed with the command line:

```
% $TRANSIMS_HOME/bin/MergeFilter <# lanes> <snapshot file name> <output file>
```

Example:

```
% $TRANSIMS_HOME/bin/MergeFilter 2 $TRANSIMS_HOME/output/Merge2/Merge2_snap merge.filtered
```

2.3.4 Left-Turn Calibrations

The number of vehicles making a left-hand turn across traffic is calibrated using the left-turn plan set. The plans for this simulation are found in *$TRANSIMS_HOME/data/plans/Left2.plans***.** Here a baseline set of vehicles traverses the circle as in the freeway or circle calibration studies. Meanwhile, vehicles turning left start at parking location 16 on link 12, cross oncoming traffic at node 1, and move onto link 11. They proceed through node 8 to link 10 and are removed from the simulation at parking location 9. The plan of *left-turn* vehicle number 1000001 is given below.

```
1000001 0 1 1 1 1
0 16 2 9 2
8628 8628 1
1 0 1
4
1000001 0
1 8
```

This microsimulation calibration is executed by running the script found in the configuration file *Left2.config*. This is executed with the command:

```
% perl $TRANSIMS_HOME/scripts/Msim.pl $TRANSIMS_HOME/config/left2_config sim.log
```

An output file is produced by filtering the vehicle snapshot data from links 7 and 11. The filtering code is in *$TRANSIMS_HOME/bin/LeftturnFilter*. It is executed with the command

```
% $TRANSIMS_HOME/bin/LeftturnFilter <# lanes> <snapshot file name> <output file>
```

Example:

```
% $TRANSIMS_HOME/bin/Leftturnfilter 2 $TRANSIMS_HOME/output/Left2/left2_snap.veh leftturn.filtered
```

This filter computes the flow of the vehicles that make the left turn to link 11 and the flow and density of the vehicles on link 7. The format of this file is shown in Table 5.

Table 5: Left-turn filter format.

Field	Interpretation
Simulation Time	Seconds since simulation start
Lane	Lane number
Flow-7	Flow of circle traffic in vehicles/hour/lane (Link 7)
Density-7	Density of circle traffic in vehicles/km/lane (Link 7)
Flow-11	Flow of left-turn vehicles in vehicles/hour/lane (Link 11)

3. TEE NETWORK

3.1 Overview

Lane changing behavior and plan following are studied using a network in the form of a "T." Vehicles start at the bottom of the "T" in three lanes and try to move to the correct lane to continue straight ahead, turn right, or make a left turn. As with the circle networks, this network is designed to test the dynamics of the Traffic Microsimulator rather than the complete TRANSIMS framework. Vehicle plan sets for traffic on this network are produced *offline*—not using the Route Planner.

The intersection at the top of the "T" is controlled by a traffic signal. Therefore, this network is also used to evaluate vehicle behavior at the signal.

3.2 Description

The Tee Network has five links and is shown in Figure 8. Links 1 and 2 are three-lane freeway type links with maximum speeds of 30 meters per second (four *cells* per second).

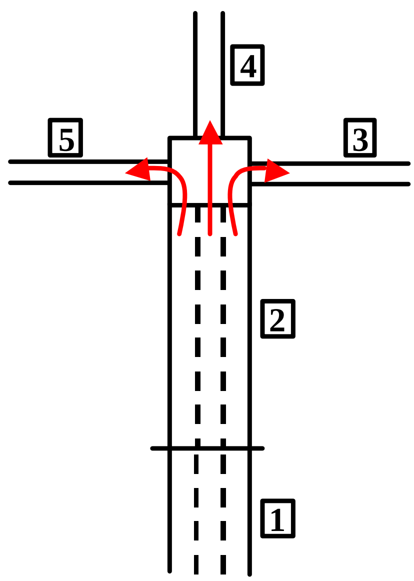

Figure 8: The five links of the Tee Network are shown. Vehicles enter on link one and exit on one of the links 3, 4, or 5.

In this network, the left lane of link 2 is connected to the single lane of link 5. The middle lane is connected to link 4, and the right one to link 3. The intersection connecting links 2 through 5 is controlled by a signal operating under various phasing cycles. The set of plans for this network has vehicles starting midway up link 2 and exiting on one of the links 3 through 5.

3.3 Usage

The tee calibration tests the weaving behavior of vehicles as they approach the intersection of links 2 to 5 at node 2. Plans to exercise this network are found in *$TRANSIMS_HOME/data/plans/Tee.plans***.** All vehicles in this plan file start at parking location 1 on link 1 and proceed through node 1 to link 2. At node 2, one-third of the vehicles, those numbered 1 to 3600, turn right to link 3; one-third, numbered 3601 to 7200, move straight ahead to link 4; and the one-third with vehicle numbers greater than 7200 turn left to link 5. The vehicles are placed on link 1 in a random lane and attempt to enter the correct lane on link 2. Three of these plans, one for each of the turning movements, follow:

```
1 0 1 1 1 1
1 1 2 3 2
1 1 1
1 0 1
4
1 0
1 2

3601 0 1 1 1 1
1 1 2 4 2
1 1 1
1 0 1
4
3601 0
1 2

7201 0 1 1 1 1
1 1 2 5 2
1 1 1
1 0 1
4
7201 0
1 2
```

The Traffic Microsimulator is executed using the configuration file *Tee_config* and the following command:

```
% perl $TRANSIMS_HOME/scripts/Msim.pl $TRANSIMS_HOME/config/Tee_config sim.log
```

Correct microsimulation behavior is determined by assessing the vehicle snapshot data for link 2. Additionally, event data will indicate any vehicle that is unable to move to the correct lane for movement through the intersection. This information is obtained by filtering the raw data. The filtering code is in *$TRANSIMS_HOME/bin/TeeFilter*. It is executed with the command

```
% $TRANSIMS_HOME/bin/TeeFilter <light cycle> <snapshot file name> <signal evolution file>
```

The light cycle time for all calibration networks given here is 60 seconds. The output from the filters is in two files. These are named *tee.lane_state.60* and *tee.vehicle_turns.60*. The format of these files is shown in Table 6 and Table 7.

Table 6: Signalized intersection filter – vehicle lane state.

Field	Interpretation
Time	Seconds since simulation start.
Box Distance From Node 2	Starting distance of box in meters measured from the node from which the vehicles are traveling away .
Light Color	State of traffic control (g=green, r=red).
Light Cycle	Length of traffic control cycle in seconds.
#in-1-to-1	Number of vehicles in the sample box during the 30-second interval that were in lane 1 and planned to turn left at the intersection.
#in-1-to-2	Number of vehicles in the sample box during the 30-second interval that were in lane 1 and planned to go straight at the intersection.
#in-1-to-3	Number of vehicles in the sample box during the 30-second interval that were in lane 1 and planned to turn right at the intersection.
#in-2-to-1	Number of vehicles in the sample box during the 30-second interval that were in lane 2 and planned to turn left at the intersection.
#in-2-to-2	Number of vehicles in the sample box during the 30-second interval that were in lane 2 and planned to go straight at the intersection.
#in-2-to-3	Number of vehicles in the sample box during the 30-second interval that were in lane 2 and planned to turn right at the intersection.
#in-3-to-1	Number of vehicles in the sample box during the 30-second interval that were in lane 3 and planned to turn left at the intersection.
#in-3-to-2	Number of vehicles in the sample box during the 30-second interval that were in lane 3 and planned to go straight at the intersection.
#in-3-to-3	Number of vehicles in the sample box during the 30-second interval that were in lane 3 and planned to turn right at the intersection.

Table 7: Signalized intersection filter – vehicle turn data.

Field	**Interpretation**
Time	Seconds since simulation start.
Light Cycle	Length of traffic control cycle in seconds.
#left turns	Number of vehicles that turned left during the 30-second interval.
#ahead	Number of vehicles that went straight during the 30-second interval.
#right turns	Number of vehicles that turned right during the 30-second interval.
#total lost	Number of vehicles that were off-plan during the 30-second interval.
#lost with plan to turn left	Number of vehicles that were off-plan and planned to turn left during the 30-second interval.
#lost with plan to go straight	Number of vehicles that were off-plan and planned to go straight during the 30-second interval.
#lost with plan to turn right	Number of vehicles that were off-plan and planned to turn right during the 30-second interval.

3.3.1 Emissions Estimator using Tee Network

The Tee Network is also used to demonstrate the Emissions Estimator. The Emissions Estimator uses Traffic Microsimulator velocity summaries to generate emissions estimates for the links in the transportation network. The Traffic Microsimulation output must be postprocessed before it is used by the Emissions Estimator. The program *TRANSIMS_HOME/bin/Readca* is a tool that postprocesses the microsimulation velocity summary output. See Volume 2—*Software,* Part 1—

Modules, Section 5.4, for a description and instructions on how to run the Emissions Estimator and the postprocessing tool, *Readca*. Sample velocity summary data from the Tee (Plan Following) microsimulation calibration network is in *$TRANSIMS_HOME/data/samples/emissions/tee.sum.vel*. The file contains velocity summaries collected over a one-hour microsimulation run on the Tee Network. A TRANSIMS configuration file that will be used by the Emissions Estimator is *TRANSIMS_HOME/config/Emissions_default*.

To postprocess the sample data, use the command:

```
% $TRANSIMS_HOME/bin/Readca $TRANSIMS_HOME/data/samples/tee.sum.vel
```

The file *readca.out* contains the postprocessed sample data.

To run the Emissions Estimator, use the command:

```
% $TRANSIMS_HOME/bin/EmissionsEstimator $TRANSIMS_HOME/config/Emissions_default
```

The emissions data is in the file *emissions.out*.

The Output Visualizer can be used to view the data. See Volume 2—*Software*, Part 1—*Modules*, Section 5.4, for instructions on how to run the Output Visualizer on the emissions data.

4. MULTIMODE NETWORK

4.1 Overview

The Multimode Network is constructed to test mode choice and a set of feedback selectors. It is used to demonstrate the transfer of travelers from the *walk* network to either a *drive* or a *transit* network. Additionally, it allows the entire TRANSIMS framework to be exercised. This is a layered network with a roadway, walkway, and a transit layer. The roadway and walk layers each form a circle with, for display purposes, the walk network set considerably inside the drive network. This leads to some non-realistic transfer times (walk times) between the walk and the other layers of the network, but does not interfere with the main purpose of this network.

The simplified population and activity generators may be used to generate activity sets on this network. The Simplified Population Generator produces single-person households that are located on the *walk* layer of the network. The Simplified Activity Generator assigns three activities to each person in the household. These are a sequence of home-work-home activities, where the work location is determined using the land-use data assigned to the activity location.

4.2 Description

There are three network layers in the Multimode Network as shown in Figure 9.

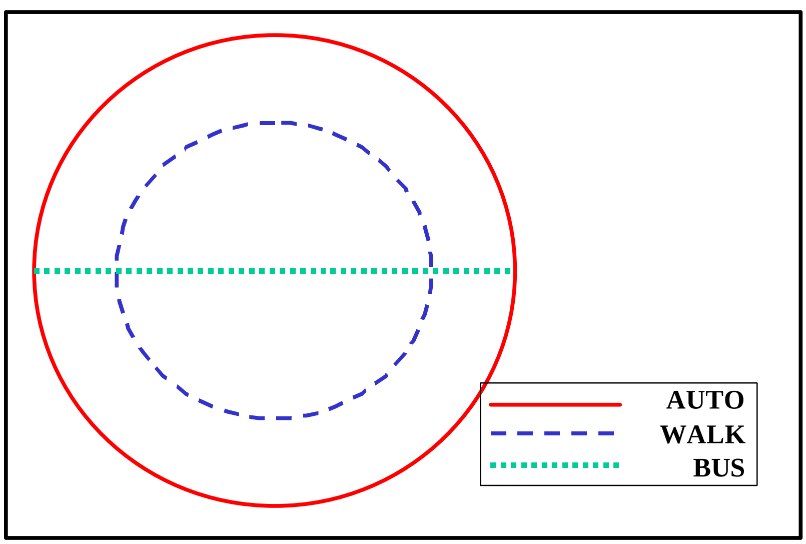

Figure 9: The Multimode Network has three layers. The auto and walk networks are in a circle. A bus-only layer runs across the circle. The layers are connected by process links.

There are 100 links on both the auto and the walk networks. The link numbers on the auto network are 1 through 100. The walk links are numbered 1001 through 1100. Each of these links has either a parking location (auto links) or an activity location (walk links). The parking location numbers are 1 through 100, corresponding to the auto link numbers. Similarly, the activity locations are numbered 1001 through 1100. There are three links on the bus network and these are numbered 10001 through 10003.

The driving or roadway links are each 750 meters (100 *cells*) in length. They are one lane in both directions and have a maximum speed of 37.5 meters (five *cells*) per second. The walk links are each 375.9 meters in length. The bus links are bidirectional one-lane roads that allow for bus traffic only. The links 10001 and 10003 are 76 meters in length, while link 10002 is 23,724.2 meters.

The bus links are given in Figure 10. These bidirectional links are restricted to buses only, and buses run on a five-minute schedule in both directions. Buses originate from the bus yards (Yard-3003 and yard-3100) on links 10001 and 10003. Their routes cover only the three links 10001, 10002 and 10003. Each bus makes two stops to load and unload passangers at the bus stops, Stop-3002 and Stop-3001 or Stop-3101 and Stop-3102.

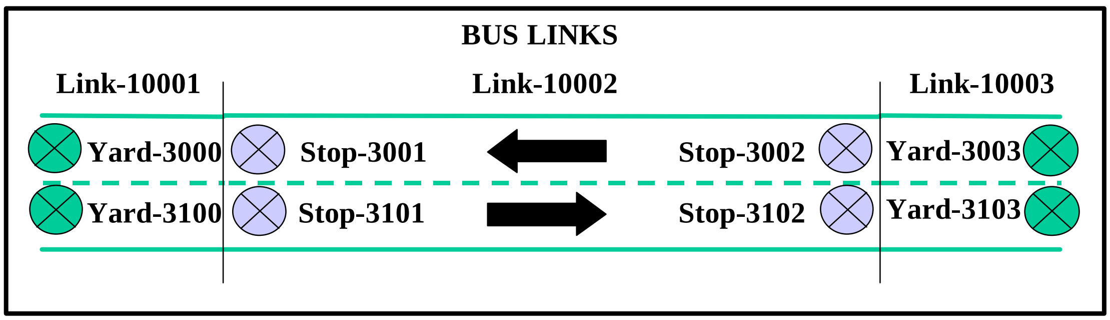

Figure 10: The three bidirectional bus links for the Multimode Network. Buses start and end at the yards and stop to load and unload passengers at the two bus stops on link 10002.

Each activity location is connected to the corresponding parking location (e.g., parking location 2 is connected to activity location 1002) by two process links. One process link allows travelers to move from the activity location to the parking location, while the other allows movement in the opposite direction. This is shown in Figure 11.

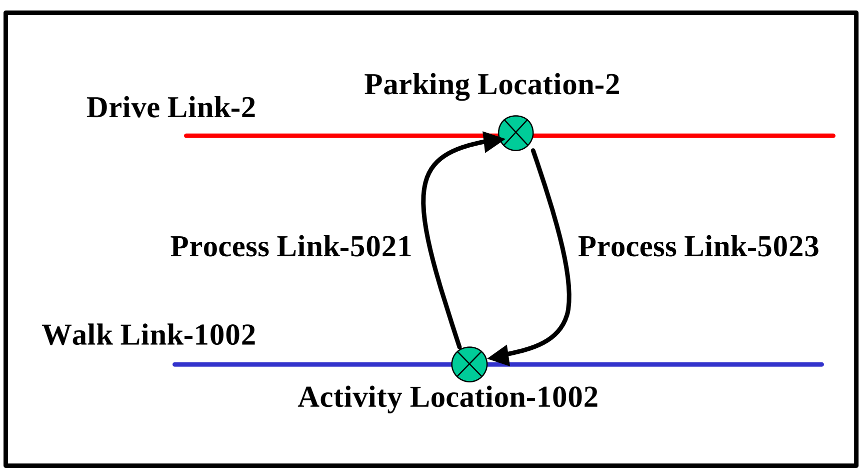

Figure 11: The process links connecting activity location 1002 to parking location 2.

The process link numbers to connect parking location p to activity location, 1000+p, are computed as:

$$5000+10*p+1 \text{ for moving from activity to parking.}$$
$$5000+10*p+3 \text{ for moving from parking to activity.}$$

Transit stops are connected to selected activity locations on the walk network by a series of process links. Figure 12 shows the process links that connect transit Stop-3001 and Stop-3002 to the activity locations 1050 and 1051. A similar set of process links connects activity locations 1001 and 1100 to bus stops, Stop-3101 and Stop-3102.

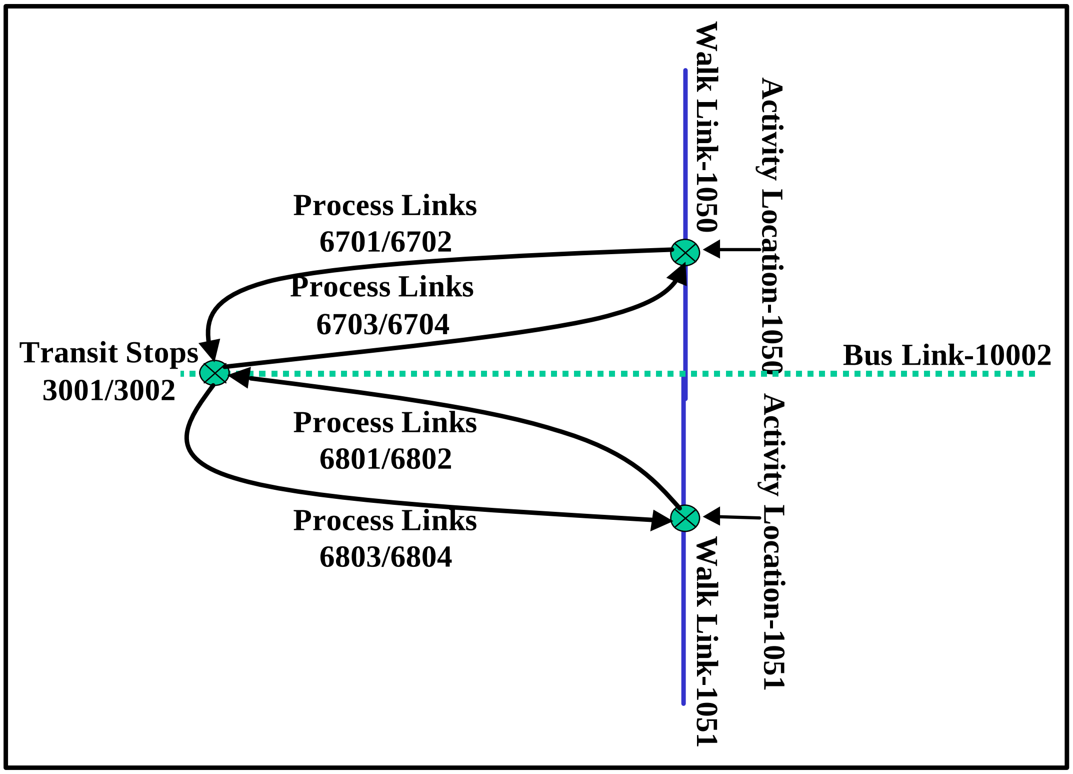

Figure 12: Process links connect the transit stops with activity locations on the walk network.

All activity locations on this network are on the *walk* layer. Three land-use variables are associated with each activity location. These are A, access to transit; H, a weight for placing households at the location; and W, a weight for work locations. Each value of A gives the walking distance in meters to the nearest transit stop. The values of W and H alternate between the values 0 and 1. Odd activity locations have H set to 1 and W set to 0. Even activity locations have W set to 1 and H set to 0.

4.3 Usage

The Multimode Network is used to demonstrate feedback through the TRANSIMS components. The Simplified Population Generator, Simplified Activity Generator, Route Planner, and Traffic Microsimulator are used in the iterations. The first iteration through the components will generate the following if they do not exist:

- synthetic population composed of single-person households located on the Multimode Network
- a TRANSIMS vehicle file for the population
- home-work-home activities for the population
- route plans for the activities
- results of the execution of the route plans in the Traffic Microsimulator

Refer to Section 4.2 for a description and diagram of the Multimode Network. Refer to Volume 2—*Software*, Part 2—*Selectors* for a description of the feedback process, iteration database, and the use of selectors that will be demonstrated on this network.

The feedback process is initiated and controlled by an iteration script. A sample script is supplied with this distribution in *$TRANSIMS_HOME/scripts/expt.sh*. This script will be used to execute TRANSIMS modules on the Multimode Network.

Refer to Volume 2—*Software*, Part 2—*Selectors*, Section 6, for a description of the functionality of the iteration script.

4.3.1 Running the Iteration Script

The iteration script is given the name of a TRANSIMS configuration file as a command line argument. The configuration file for the Multimode Network is *$TRANSIMS_HOME/config/Calnet_default*.

Select a run directory. RUN_DIRECTORY in the following instructions refers to the complete pathname of this directory. The output from the feedback iterations requires ~20 megabytes of disk space in the run directory and ~300 megabytes of space in the *TRANSIMS_HOME/output/Calnet directory*. You must also have read and write permissions in both directories.

Change directory to run directory.

```
% cd <RUN_DIRECTORY>
```

Execute the iteration script from the run directory using the Multimode configuration file, *$TRANSIMS_HOME/config/Calnet_default*.

```
% /bin/sh $TRANSIMS_HOME/scripts/expt.sh $TRANSIMS_HOME/config/Calnet_default
```

The iteration script will place information about the iterations in the RUN_DIRECTORY. It will create a subdirectory for each iteration(it.0, it.1, …). Three iterations are specified in the *Calnet_default* configuration file.

Iteration 0 creates a population, home-work-home activities for the population, and route plans for the activities. The execution of the route plans in the Traffic Microsimulator produces information about each traveler’s trips. The Selector places pertinent information about the traveler into an iteration database. This information is used to select travelers for activity or route plan changes in future iterations. Subsequent iterations use the selected travelers to regenerate activities, make new route plans, and then execute the plans in the Traffic Microsimulator.

Volume 2—*Software*, Part 1—*Modules*, Section 6.3, has instructions on how to run the Output Visualizer to view output of the TRANSIMS modules.

5. LOCAL STREETS NETWORK

5.1 Overview

The Local Streets Network allows all of the modules of TRANSIMS to be tested. The purpose of this network is to demonstrate the usage of the full TRANSIMS Population Synthesizer and Activity Generator modules and the construction of a more complex network. Additionally, with particular selector scripts, the effects of local streets on traffic behavior may be tested. The network consists of a grid of local streets surrounded by a set of arterial streets and freeways. The local streets contain both a walk layer and a street network. Activity locations are on both the walk layer and the major roads.

5.2 Description

The Local Streets Network is pictured in Figure 13. It consists of a grid of local streets surrounded by arterial roads. Major roads feed this system of local streets, one from each direction. The link numbers of the major road system are given in the boxes in Figure 13, while the four major intersection at nodes, 17 through 20, are identified in the circles.

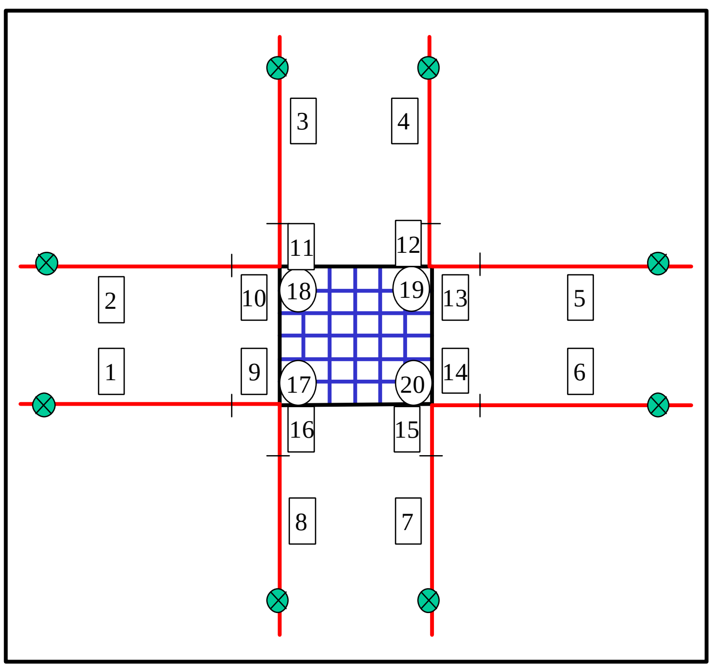

Figure 13: The Local Streets Network showing both the local street grid and the major roads leading into the area. Link numbers are given in the boxes, while the major intersections at nodes 17 through 20 are denoted by circles.

All links in this network are bidirectional. Links 1 to 16 shown in Figure 14 contain two lanes in each direction. They are restricted to vehicular traffic only. The maximum speed on links 1 through 8 is 37.5 meters per second (five *cells* per second), while the maximum speed on links 9 through 16 is 30.0 meters per second (four *cells* per second). Links 1 through 8 are connected to links 9 through 16 by *seamless* nodes—that is, vehicles pass through these nodes without pausing.

The circles with crosses on links 1 through 8 denote activity and parking locations. Links 9 through 16 do not have either activity or parking locations. In contrast to the Multimode Network, where all activity locations were on the walk network, the activity locations on links 1 through 8 are placed on the roadway network. Activity locations like these at the boundaries of the network and on the vehicular layer are common in networks for metropolitan areas. They are the destinations and origins of travelers entering and/or leaving the area under study. The activity and parking locations on these links are connected by process links.

In this network, links contain at most one parking and activity location. Each link with a parking location has an associated activity location, and they are connected by process links. The parking locations have the same number as the links. The activity locations have numbers of 1000 plus the link number. Process links that allow travelers to move from the parking location to the associated activity location are numbered as 10,000 plus the link number. The process links for movements in the opposite direction are numbered 20,000 plus the link number.

The link numbers of the major arterial streets surrounding the local streets are shown in Figure 14. The nodes connecting these links are 51-55, 61-65, 71-75 and 81-85. These arterials are two lanes in both directions, and both left- and right-hand turns are allowed from the arterials into the local streets. Yield signs control movement from the local streets to the arterial network.

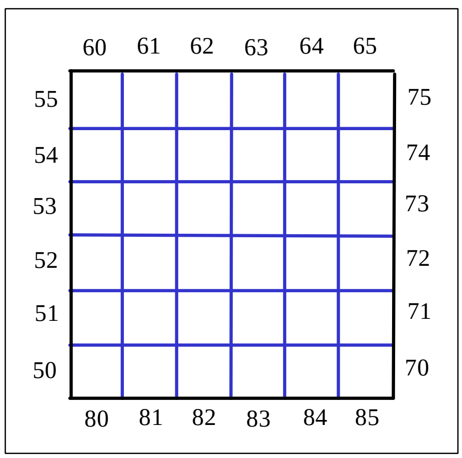

Figure 14: The local street grid is connected to a series of two-lane bidirectional arterial links. Left- and right-hand turns area allowed from the arterials into the local streets. Egress from the local streets to the arterial network is controlled by yield signs.

Local streets are numbered from South to North and West to East. In the South-North direction, the first series of links on the left is numbered 111 through 116, the next series is 121-126, and the last is 151-156. In the West-East direction, the links to the South are 211 through 216, while those to the North are 251-256. The series of links 131-136 and 231-236 are through streets with no controls except where they meet. Here there is a four-way stop. All other intersections on the local street grid are controlled by four-way stops.

The intersections at nodes 17-20 are signalized and each contains a turn pocket. The turn movements from one direction at each of these nodes are shown in Figure 15.

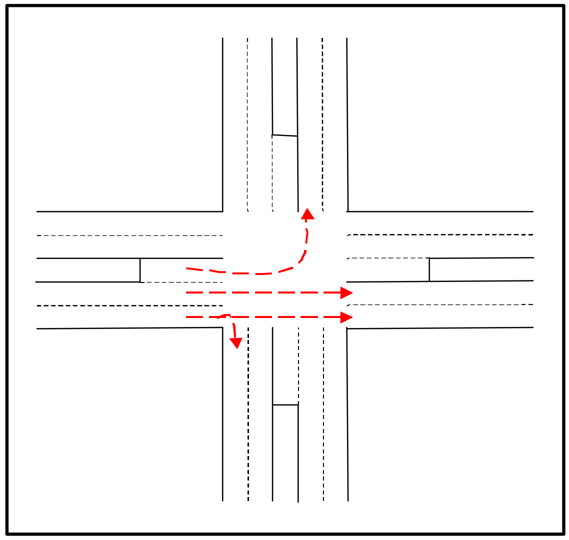

Figure 15: This diagram shows the turning movements at the major intersections at nodes 17 and 20. These are signalized intersections, and the turn movements are symmetric around the node.

Associated with the activity locations is a set of land use data that is typical of a metropolitan region. In addition to the required characteristics, these are:

1) TAZ, the traffic analysis zone

2) TRACT, the census tract

3) BLOCKGR, the census block group

4) Com.area, the area in square feet of commercial property surrounding the activity location

5) Ind.area, the square footage of industrial area

6) Mfr.area, the square footage of multifamily residential area around the link

7) Pos.area, the square footage of public land around the activity location

8) Rur.area, the square footage of rural area around the link

9) Sfr.area, the square footage of single family residence assigned to the activity location

10) employees, the number of employees associated with the activity location

11) employers, the number of employers at the activity location

The activity locations on the first eight links and shown in Figure 13 are the industrial, manufacturing, and commercial locations on the network. Each has nonzero entries for Com.area, Ind.area, employees, and employers. The individual activity locations are each assigned a unique TAZ from 1 through 8. The TRACT and BLOCKGR are set to 0 for these locations. The activity locations around the border of the local streets are all assigned to TAZ 9. All other entries are zero for these locations. The activity locations on the local streets are all in TAZ 10, TRACT 1, and

BLOCKGR 2. These are residential locations, and there are positive entries for Mfr.area and Sfr.area. All other land-use variables have the value zero.

5.3 Usage

The Local Streets Network demonstrates the major modules of TRANSIMS. The programs are located in the *TRANSIMS_HOME/bin* directory. A TRANSIMS configuration file is required by most of the TRANSIMS modules. Sample configuration files for the Local Streets Network are *TRANSIMS_HOME/config/Local_config1*, *Local_config2*.

Select a run directory and invoke the following TRANSIMS modules from this directory. *RUN_DIRECTORY* in the following instructions refers to the complete pathname of this directory. Approximately 250 megabytes of output will be produced, so choose a run directory on a file system that has enough free space. You must have read and write permissions in the run directory and in the *TRANSIMS_HOME/output/Local* directory.

Change directory to the run directory.

```
% cd <RUN_DIRECTORY>
```

Copy the sample configuration files for the Local Streets Network, *$TRANSIMS_HOME/config/Local_config1* and *$TRANSIMS_HOME/config/Local_config2*, into the run directory.

```
% cp $TRANSIMS_HOME/config/Local_config1 <RUN DIRECTORY>

% cp $TRANSIMS_HOME/config/Local_config2 <RUN DIRECTORY>
```

Edit the local copy (in your run directory) of *Local_config2*. Change the entries for *VEHICLE_FILE* and *PLAN_FILE* by inserting your run directory name between the < > brackets, then remove the < > brackets.

Copy the sample vehicle file for the Local Streets Network (*$TRANSIMS_HOME/data/vehicles/local.vehicles*) into your run directory.

```
% cp $TRANSIMS_HOME/data/vehicles/local.vehicles <RUN DIRECTORY>
```

Change the file permissions to read and write on the *local.vehicles* file.

```
% chmod u+rw ./local.vehicles
```

The TRANSIMS modules must be run in the following order:

1) **Population Synthesizer** (*TRANSIMS_HOME/bin/Syn*)
   A baseline synthetic population will be generated from sample census data supplied with this release. PUMA00400 from New Mexico census data is used to generate the population. Refer to Volume 2—*Software*, Part 1—*Modules*, Sections 1.3 and 1.4 for complete instructions (including setting the necessary environment variables) on running the Population Synthesizer. Generation of this small population demonstrates the process by which large populations are created from census data.

   Example:

```
% $TRANSIMS_HOME/bin/Syn
```

The Population Synthesizer creates three population files:

Family Population <prefix>Family_Synthetic_HHRecs.out
Non-Family Population <prefix>Non_Family_Synthetic_HHRecs.out
Group Quarters Population <prefix>Group_Synthetic_HHRecs.out

<prefix> refers to the string that you typed into the Population Synthesizer text entry window. This string is prepended to the names of the population files.

These files are combined manually to make a single synthetic population file. Edit the nonfamily *(<prefix>Non_Family_Synthetic_HHRecs.out*) and group quarters *(<prefix>Group_Synthetic_HHRecs.out*) files and remove the header lines from each file. The header lines are the first two lines in each file that begin with "Household Demographics:" and "Person Demographics:", respectively.

Concatenate the three files into a single population file, *mypop.baseline*.

```
% cat <prefix>Family_Synthetic_HHRecs.out > mypop.baseline

% cat <prefix>Non_Family_Synthetic_HHRecs.out >> mypop.baseline

% cat <prefix>Group_Synthetic_HHRecs.out >> mypop.baseline
```

The combined population file, *mypop.baseline*, should contain the header lines from the family population file and the data lines from all three files. This combined baseline population file will be used to demonstrate how to locate a population on a transportation network and then how to generate a TRANSIMS vehicle file from the located population.

2) **Block Group Locator** (*TRANSIMS_HOME/bin/BlockGroupLoc*)
Each household in the baseline population will be assigned a home location on the Local Streets Network. Each household and person in the population will also be assigned a unique TRANSIMS ID. The Block Group Locator uses the TRANSIMS configuration file, *Local_config1*. The combined population file, *mypop.baseline*, is named in the configuration file. The located population will be placed in the file *mypop.located*.

```
% $TRANSIMS_HOME/bin/BlockGroupLoc Local_config1
```

3) **Vehicle Generator** (*TRANSIMS_HOME/bin/Vehgen*)
The Vehicle Generator creates a TRANSIMS vehicle file that contains an entry for each vehicle in a household. The starting location of the vehicle will be a TRANSIMS parking location that is accessible via the walk network from the household's home location. The Vehicle Generator uses the TRANSIMS configuration file *Local_config1*. The located population, *mypop.located*, is named in this configuration file. The results of the Vehicle Generator will be in the file, *mypop.vehicles*.

```
% $TRANSIMS_HOME/bin/Vehgen Local_config1
```

4) **Activity Generator** (*TRANSIMS_HOME/bin/ActivityGenerator*)
The Activity Generator generates activities for persons in the population. In this release, all modes in activities are car mode. The Activity Generator does not use a TRANSIMS configuration file; instead a special configuration file is used—

TRANSIMS_HOME/config/atp.config. The activities that are generated will be in the file *local.act*. Refer to Volume 2—*Software*, Part 1—*Modules*, Section 2 for a description of the Activity Generator.

```
% $TRANSIMS_HOME/bin/ActivityGenerator $TRANSIMS_HOME/config/atp.config
```

5) **Route Planner** (*TRANSIMS_HOME/bin/Router*)
The Route Planner creates route plans on the Local Streets network from the activities generated by the Activity Generator. Refer to Volume 2—Software, Part 1—*Modules*, Section 3 for a description of the Route Planner. The Route Planner uses the TRANSIMS configuration file *Local_config2*. The vehicle file, *local.vehicles*, will be used by the Route Planner to coordinate with the activities produced by the Activity Generator.

```
% $TRANSIMS_HOME/bin/Router Local_config2
```

6) **Traffic Microsimulator** (*TRANSIMS_HOME/bin/PVM.ARCH.LINUX/CA*)
The Traffic Microsimulator simulates the movement and interactions of travelers in the transportation system. Using a plan provided by the Route Planner, each traveler attempts to execute that plan on the transportation system. The combined traveler interactions produce emergent behaviors such as traffic congestion. Refer to Volume 2—*Software*, Part 1—*Modules*, Section 4 for a description of the Traffic Microsimulator. The microsimulation is invoked from a Perl script that initializes the communication mechanism (*TRANSIMS_HOME/scripts/Msim.pl*). You must know the location of the *perl* executable on your system. The Perl script (*Msim.pl*) requires the full path name of a TRANSIMS configuration file as a command line argument. The Traffic Microsimulator uses the TRANSIMS configuration file, *Local_config2*. The vehicle file, *local.vehicles*, will be used by the Traffic Microsimulator to coordinate with the plans produced by the Route Planner (*local.plans*).

The Traffic Microsimulator program resides in a subdirectory of *$TRANSIMS_HOME/bin*. Since the executable depends on the type of parallel communication that will be used, Parallel Virtual Machine (PVM), or Message Passing Interface (MPI), two executables are provided in the following subdirectories:

$TRANSIMS_HOME/bin/ARCH.PVM.LINUX

$TRANSIMS_HOME/bin/ARCH.MPI.LINUX.ch_p4

The user may specify the parallel communication mechanism in the TRANSIMS configuration file. PVM is the default and is used in the configuration files supplied with this release.

```
% /usr/bin/perl $TRANSIMS_HOME/scripts/Msim.pl <RUN_DIRECTORY>/Local_config2 sim.log
```

The microsimulation results will be in the *$TRANSIMS_HOME/output/Local* directory, which is specified in the configuration file using the key, *OUT_DIRECTORY*. The simulation log file is *sim.log* in your run directory.

7) **Output Visualizer** (*TRANSIMS_HOME/bin/Vis*)
Refer to Volume 2—*Software*, Part 1—*Modules*, Section 6 for a description and instructions on how to use the Output Visualizer to display the output of the Traffic Microsimulator and Route Planner. The Output Visualizer uses the same configuration file that was used by the Traffic Microsimulator, *RUN_DIRECTORY/Local_config2*. The output of the Traffic

Microsimulator is in the TRANSIMS_HOME/output/Local directory. The name of the vehicle snapshot file is *local_snap.veh*.

```
% $TRANSIMS_HOME/bin/vehtobin $TRANSIMS_HOME/output/Local/local_snap.veh local_snap.veh.bin
```

```
% $TRANSIMS_HOME/bin/Vis Local_config2
```